

Megawin 8051 OCD ICE 使用说明书(简体中文版)

购买信息: <u>http://item.taobao.com/item.htm?id=9368267707</u>



目录

1	介绍	. 3
	特色	. 3
	描述	. 3
2	硬件配置	4
3	软件设定	. 5
	3.1 安装 ICE 转接器的驱动程序	. 5
	3.2 新增 Megawin 芯片的资料到 Keil 8051 IDE	. 5
4	Keil IDE 设定	6
	4.1 Device 选项	. 7
	4.2 Target 选项	. 7
	4.3 Output 选项	. 8
	4.4 C51 选项	. 8
	4.5 Debug 选项	. 9
_	4.6 Utilities 选项	10
5	开始调试	11
	5.1 启动 dScope-Debugger 功能	11
	5.2 介绍调试坏境	12
	5.2.1 复位(Reset)/执行(Run)/停止(Halt)/平步(Step)/执行到呆行(Run-to-Cursor)	13
	5.2.3 设定断占	13 14
	5.2.4 显示/编辑外围寄存器的内容	15
	5.2.5 检视反编译窗口	16
	5.2.6 检视查看窗口	17
	5.2.7 检视内存窗口	18
6	注意事项	19
	6.1 寄存器定义文件	19
	6.2 内建 XRAM 及外部数据存储器	19
	6.3 程序代码优化及原始码调试	20
	6.4 for 循环的原始码调试	21
	6.5 使用调试时的硬件选项要求	21
	6.6 错误讯息	22
	6.7 正确的连接 ICE 仿真器到计算机	23
修	订记录	24
		•



<u>1 介绍</u>

特色

- * 笙泉专利的 OCD(On-Chip-Debug) 技术
- * MCU 内建实时调试
- * 独立的两个接脚串列接口,所有1/0口可仿真,不占用系统的接脚,通过OCD_SCL,OCD_SDA引脚仿真(原ALE,PSEN引脚)
- * 直接兼容于 Keil 的 8051 IDE 调试仿真接口
- * 使用 USB 连接计算机于系统
- * 强大的调试动作:复位、全速执行、暂停、单步执行...等等
- * 可程序化的断点,可以同时插入四个断点
- * 多个有用的调试窗口: 寄存器/反编译/监看变量/内存 窗口

<u> 描述</u>

这个全新的"Megawin 8051 OCD ICE"对 8051 嵌入式系统来说是一个强而有力的开发工具。他是采用笙泉科技的 OCD(On-Chip-Debug)专利技术,这个 ICE 提供内建实时调试的功能。在开发调试时使用者不需要像传统的 8051 ICE 一样,准备任何的开发板或者是转接脚座即可以开发调试。使用者唯一需要做的就是预留一个 4 只脚位的连接 器给专属的 OCD 接口即可: VCC, OCD_SDA, OCD_SCL 以及 GND。[TH065C需要留6个引脚,详细见下表]

另外,它最有用的特色是能够直接连接用户的系统到 Keil 8051 IDE software 的界面做调试,而它是直接使用 Keil IDE 的 dScope-Debugger 功能来做调试并且承接了所有 Keil 的优点。

注:

"Keil"是 "Keil Elektronik Gmbh and Keil Software, Inc."的注册商标,而 "Keil 8051 IDE software"是8051 嵌入式系统的开发环境中最普遍的一个。

"Megawin 8051 OCD ICE"的ICE转接器的照片(TH065B)



This document information is the intellectual property of Megawin Technology Co., Ltd. © Megawin Technology Co., Ltd. 2010 All right reserved.



<u>2 硬件配置</u>

当要做调试时,使用者必须使用 ICE 转接器将计算机与系统连接起来,如下图。ICE 转接器是使用 USB 的电源,因此使用 ICE 时是不需要再接其他的电源的。

硬件连接图



注:更多信息请參考 <u>6.5 节</u>。

OCD ICE界面的脚位编号

Part No.	Package	OCD_SCL	OCD_SDA	RST	CLK
	40-pin DIP	29	30	N/A	N/A
MPC82G516	44-pin PLCC	32	33	N/A	N/A
	44-pin QFP	26	27	N/A	N/A
	44-pin QFP	26	29	4	5
WG62FL(E)552/564	48-pin LQFP	28	32	5	6
MC84EC516	48-pin LQFP	26	27	25	N/A
WG04FG516	64-pin LQFP	34	35	33	N/A

**N/A:不需连接



<u>3 软件设定</u>

这个章节会告诉你在使用 OCD ICE 之前要如何做软件的设定。

3.1 安装ICE转接器的驱动程序

使用者只需要把这个 ICE 转接器直接插上任何一个 USB 端口就可以了,不需要安装任何驱动程序。

3.2 新增Megawin芯片的资料到Keil 8051 IDE

首先, 先将 ICE 转接器插到计算机的 USB 端口, 然后执行目录[Database Installer]里的"Setup.exe"将 Megawin 的芯片信息 新增至 Keil 8051 IDE 内。当然, 您可以新增到 Keil 8051 IDE 的 µVision2 / µVision3 / µVision4 都可以。

在打开 Database Installer 后,请依照下列顺序完成新增动作,如图示。(Keil μVision4已集成,以下2个步骤非必要的)

步骤一:按下 Browse 钮指定 Keil 的安装目录。 (一般来说,安装 Keil 8051 IDE 时预设的安装路径为"C:\KEIL".必须和Keil目录相同,否则调试时会提示错误信息。)

步骤二: 按下 Install 钮开始新增 Megawin 芯片的资料到 Keil 内。

安装过程图解





<u>4 Keil IDE 设定</u>

在使用 Keil IDE 的 dScope-Debugger 功能之前用户必须先对 Keil IDE 做一些设定。首先,先开启您要调试的 μVision 项目,然后在"Target-.."的地方按下右键并点选"Options for Target"如下图所示:

🕎 test - 猩ision3 - [C:\tmp\test\main.C]		. ð 🗙
Eile Edit View Project Debug Flash Peripherals Click right key Image: Copions for Target - test Image: Copions for Target 'Target- test' Image: Copions for Target Build target Image: Components Remove Item Image: Components Image: Components Remove Item Image: Components Image: Components Image: Components Image: Components Image: Components Image: Components Image: Click Time Image: Click Time	<pre>Tools SVCS Window Help Tools SVCS Window Help M # + + M # + + M # @ M M M M M M M M M M M M M M M M M M M</pre>	
Build (Command) Find in Files /		•

Set Project options

Megawin On-Chip-Debug Driver L:1 C:1 R/W



4.1 **Device** 选项

选择"Megawin Device Database"以及型号

仿真环境下不要选"Generic CPU Data Base" Options for Target 'Target- test' BL51 Locate BL51 Misc Debug Utilities Device Target Output Listing C51 A51 Database: Megawin Device Database • Vendor: Megawin Megawin Device Databas Device: MPC82G516 Use Extended Linker (LX51) instead of BL51 Toolset: C51 Use Extended Assembler (AX51) instead of A51 (1) Operating speed up to 24MHz
(2) 64K bytes Flash program memory
(3) 256 bytes internal RAM 🖃 🔷 Megawin MPC82G516 (4) 1024 bytes on-chip eXpanded RAM (XRAM)
 (5) Four full I/O ports P0/P1/P2/P3, and half P4
 (6) Three 16-bit Timers (7) Programmable clock output
 (8) Four-level priority interrupt system (9) Two UARTs (10) Eight keypad interrupt inputs (11) One-time enabled Watch-dog Timer (12) SPI (Serial Peripheral Interface) (13) PCA (Programmable Counter Array) with 6 channels (14) 10-bit ADC with 8 multiplexed analog inputs
 (15) ISP (In-System Programming) & ICP (In-Circuit Programming)
 (16) IAP (In-Application Programming) memory with programmable size (17) OCD interface for on-chip debugging

取消

Defaults

4.2 Target 选项

勾选"Use on-chip ROM"以及"Use on-chip XRAM"。

Options for Target 'Target- test'
Device Target Output Listing C51 A51 BL51 Locate BL51 Misc Debug Utilities
Megawin MPC82G516 Xtal (MHz): 12.0 Memory Model: Small: variables in DATA Code Rom Size: Large: 64K program Operating system: None
Off-chip Code memory Start: Size: Off-chip Xdata memory Eprom #1:
Code Banking Start: End: Banks: 2 Bank Area: 0x0000 0xFFFF
確定 取消 Defaults 説明

This document information is the intellectual property of Megawin Technology Co., Ltd. © Megawin Technology Co., Ltd. 2010 All right reserved.

確定

說明



TH065B+ User Manual, v2.40

4.3 Output 选项

勾选"Debug Information"。这个选项必须勾选才能够产生出 ICE 调试所需的 OMF(Object Module Format)档案。

Options for Target 'Target- test'	×
Device Target Output Listing C51 A51 BL51 Locate BL51 Misc Debug Utilities Select Folder for Objects Name of Executable: test	
 Create Executable: .\test Enabled Debug Information Create HEX File HEX Format: HEX-80 	
C Create Library: .\test.LIB	æ Batch File
🔽 Beep When Complete 🔽 Start Debugging	
Run User Program #1: test.BAT	Browse
Run User Program #2:	Browse
確定 取消 Defaults	說明

<u>4.4 C51 选项</u>

选择"Level 0: Constant folding"用以关闭程序代码的优化。详细资料请参考 6.3 节。注:这个设定是非必要的。

Options for Target 'Target- te	est'	
Device Target Output L Preprocessor Symbols Define: Undefine: Code Optimization -	Listing C51 A51 BL51 Locate	BL51 Misc Debug Utilities Warnings: Warninglevel 2
Level: 0: Constant fol Emphasis: 0: Constant fol Emphasis: 1: Dead code e. 2: Data overlay 3: Peephole op 4: Register var 5: Common su 6: Loop rotatic	ding ding ding ding ding ding ding ding	Bits to round for float compare: 3 Interrupt vectors at address: 0x0000 Keep variables in order Enable ANSI integer promotion rules
Include Paths Misc Controls Compiler control string	SPEED) BROWSE DEBUG OBJECTEXT	END CODE
	確定	Defaults



TH065B+ User Manual, v2.40

4.5 Debug 选项

选择"Megawin On-Chip-Debug Driver"。

Device Target Output Listing C51 A51 BL51 Output Listing C51 A51 BL51 Limit Speed to Real-Time Select Megar Load Application at Startup IV Run to main() Initialization File: Restore Debug Session Settings Vertication Settings Verticati	Locate BL51 Misc Debug Utilities Use: Megawin On-Chip-Debug Driver Settings Win Keil Monitor-51 Driver Keil ISD51 In-System Debugger MON390: Dallas Contiguous Mode LPC900 EPM Emulator/Programmer Initializatic ST-uPSD ULINK Driver ADI Monitor Driver Silicon Laboratories C8051Fxxx uVision Megawin On-Chip-Debug Driver Silicon Laboratories C8051Fxxx uVision Megawin On-Chip-Debug Driver Silicon Laboratories C8051Fxxx uVision Megawin On-Chip-Debug Driver Watchpoints Memory Display Driver DLL: Parameter:
Dialog DLL: Parameter: DP51.DLL -p52 確定 耳	Dialog DLL: Parameter: MegawinOCD.DL -pBA122 又消 Defaults 説明

并且勾选"Load Application at Startup"以及 Cache Options 里的所有选项。

Device Target Output Listing C51 A51	BL51 Locate BL51 Misc Debug Utilities Settings
Limit Speed to Real-Time	Enabled
✓ Load Application at Startup ✓ Run to main() Initialization File:	✓ Load Application at Startup ✓ Run to main() Initialization File:
Edi	t Edit
Restore Debug : Target Setup Cache Options Watchpoin Memory I Cache Data CPU DLL: S8051.DLL Cache Code Enabled	Restore Debug Session Settings Breakpoints Watchpoints Memory Display Driver DLL: Parameter: S8051.DLL
Dialog DLL: Parameter: DP51.DLL -p52	Dialog DLL: Parameter: MegawinOCD.DL -pBA122



4.6 Utilities 选项

"Update Target before Debugging"一定要关闭,因为我们已经勾选了"Load Application at Startup"参考<u>4.5</u>节。 而"Use Target Driver for Flash Programming"可以乎略不用管它。

主: mVision2 没有这个选项。	1 这两个选项不要选择,让它空着
Options for Target 'Target- test'	×
Device Target Output Listing C51 A51 BL51 Locate BL51 Misc Configure Flash Menu Command Use Target Driver for Flash Programming Silicon Laboratories C8051Fxxx uVision Settings Init File: Use External Tool for Flash Programming Command: Arguments: Run Independent	Debug Utilities
	ults



<u>5 开始调试</u>

当您完成第二、三、四节的设定后,您就可以开始使用 µVision 来做调试的功能了。

5.1 启动 dScope-Debugger 功能

在做完项目的设定后(假设没有错误的话),您就可以按下 dScope 钮进入 Keil IDE 的调试模式了,按下后会自动将您的程序下载到 MCU 内部,而这个过程会花一点时间。

🌃 test - 猩ision3 - [C:\tmp\test\main.0			BX
📄 Eile Edit View Project Debug Flash	Peripherals Iools SVCS Window Help	dScope Button	BX
🏠 😅 🖬 🕼 👗 🛍 🗎 요 오 🎼	■ ふ% % % 🙀 🔽 🛤 🕂	+ → (2) ≤ (2) E B (2) ← → (2)	
🕸 🕮 🎽 🛱 🛣 Target-test	🗾 🏝 📟	Start/Stop Debug Session	
Project Workspace v x	001 - /***********************************	*********	
E Target- test	002 * main.C 003 ***********************************	******	, –
Startup A51	004		
Hain.C	005 #include "REG_MPC82G516.h" 006 #include "delay.h"	and button to start debugging	
Delay_1T.A51	007 CIICK GS	scope button to start debugging	
E Include Files	008 009 #define LED Y P12 //		
Delay.H	010 #define LED_G P13 //ureen		
	012 #define LED_0 P15 //Orange		
	013 014 word show MCIL rup(word):		
	015 void LED_Y_on(void);		
	016 void LED_Y_off(void);		
	018 void LED_G_off(void);		
	019 void LED_R_on(void); 020 void LED_R_off(void);		
	021 void LED_O_on(void);		
	022 void LED_U_off(void); 023		
	024 void test_SFR(void);		
	025 V01d Teset_SFR(V01d); , 026		-
🖹 🔛 B 🕅 F	🗎 main.C		
* All rights reserved.			
Status: Creating binary fil		10.1.1.1.1.1.0.00000000	
a dress 0x000002C9. Stat	Ing 12 bytes at address UXUUUUUUUb. DO2D9. Status: Writing 16 bytes at	addre	
g Status: HEX to BIN conversi	n was successful.		
<pre>s "test" - U Error(s), O Warn </pre>	ng(s).		-
8 K K Build & Command & F	id in Files /		
	Megawin On-Chip-Debug Driver	L:8 C:40	R/W



<u>5.2 介绍调试环境</u>

在调试的环境里可以看到有四个基本的窗口,他们分别是寄存器(Register)窗口、反编译(Disassembly)窗口、监看 变量(Watch)窗口及内存(Memory)窗口,详细说明如下:

寄存器窗口

这个窗口会显示出目前的寄存器值(R0~R7),还有系统寄存器(A,B,SP,DTPR 及 PC)还有程序狀态字符(PSW)。当 寄存器显示为蓝底时代表他正被目前的指令改变他的数值。

反编译窗口

这个窗口在进入调试模式后就会自动打开,他会将目前的程序代码以相对应的汇编语言显示出来。

查看窗口

当目前选在 Locals 分页时,这个窗口会自动将区域变量显示出来。这个区域变量的值包括在主循环 main()的变数。如果要查看全局的变量则必须先将分页选到 Watch #1 或 Watch #2,然后按下<F2>并输入您要查看的变量名称即可,同样的,当变量为蓝底时代表他被目前的指令改变量值。

内存窗口

这个窗口可以显示 **data/idata/xdata/code** 内存空间的内容,可以使用的命令为 **d:0x00~d:0xFF**, **i:0x00~i:0xFF**, **x:0x0000~x:0xFFFF** 以及 **c:0x0000~c:0xFFFF**,使用者可以用相对应该的命令查看这四种内存的内容。

😵 test - 猩ision3 - [Disassembly]		×
😤 Eile Edit View Project Debug Flash Perip	pherals Iools SVCS Window Help	×
🏠 🚔 🖬 🎒 🍐 🏷 🛍 🗎 의 으 🛛 (東 東 ス	اﷺ ﷺ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦ ♦	
🛔 🗿 🗄 😯 🖓 🖓 🖓	2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
Project Workspace 🔹 🗙	31: void main(void)	~
Register Value	32: { Disassembly	5
⊡Regs	33: char L_var1,L_var2,L_var3; //local variables Window	
10 0x00	35:	
	36: char xdata X_var1,X_var2; //local variables, in 'xdata' memory spa	
r3 0x00 Register	37: 38: LED blinking():	
r4 0x00 Window	39:	
r5 0x00	C:0x02EF 120357 LCALL LED_blinking(C:0357)	
rb UxUU	C:0x02F2 75085A MOV 0x08,#0x5A	
E-Svs	41: L_var2=0x5B;	
a 0x00	C:0x02F5 75095B MOV 0x09,#0x5B	
Ъ 0х00	C:0x02F8 750A5C MOV 0x0A,#0x5C	
sp 0x13	43: L_var4=0x1234;	
PC \$ 0x02ef	C:0x02FB 750B12 MOV 0x0B,#0x12 C:0x02FE 750C34 MOV 0x0C,#0x34	
	44: L_var5=0x5678;	~
🖹 Files 🗮 Regs 🔑 B 📢 Fu 🔍 Te	45.	×
Symbols • ×	E Main.C 🕵 Disasse	
* Load "C:\\tmp\\test\\test"	Name Value Address: x:0x0000	•
WS 1, G_Var1 WS 1, `G var2	L_var1 0x00 X:0x000000: 00 00 DF E0 6B 71 77 FD	
	L_Vai2 0x00 X:0x000008: BB 73 5C 38	
	L_{var4} Window 0x0000 X:0x000010: DC Window 1D 31 C7	
	L_var5 0x0000 X:0x000020: F6 EA AA 5B	
>	-X_var1 0x00 X:0x000028: CD 19 39 CF F7 1B F9 49	
ASM ASSIGN BreakDisable	x_var2 UxUU (X:0x000030: D7 AB D4 65 7F DB 77 B7	-1
BHI + H Command A + +	Memory #1 / Memory #2 / Memory #	
Ready	Megawin On-Chip-Debug Driver t1: 0.00000000 sec	W



5.2.1 复位(Reset)/执行(Run)/停止(Halt)/单步(Step)/执行到某行(Run-to-Cursor)

复位、执行、停止、单步及执行到某行是基本的调试动作,使用者可以轻易的在 GUI 的快捷列中执行这些功能,如下图:

🌄 test - 猩isio	n3 - [Disassembl y]		_ = 🛛 🔀
🕂 Eile Edit Vi	ew <u>P</u> roject <u>D</u> ebug Fl <u>a</u> s	h Peripherals Iools SVCS Window Help	_ & ×
12 🖨 🖬 🕼	1 B B 2 2 4	「幸んぷぷぁ┓」	
RST 🗄 🖾 🖓	() () *() ⇒ ½± 0±		
Project Workspace	- ×	36: void main(void)	~
Register	Reset/Run/Halt/S	Step : {	
🖻 Regs		: //declare local variables	
r0	UXUU	: unsigned char L_var1,L_var2,L_var3;	
r1	0x00	40: unsigned int L_var4,L_var5;	
r2	0x00	41: unsigned char xdata X_Var1,X_Var2; //in 'xdata' space	
	0x00	42: unsigned char 1;	
r4	0x00	$44 \cdot (/\text{FD blinking}) \cdot$	
r5	0x00	45:	
тб	0x00	46: L_var1=0x5A;	
r7	0x00	C:0x02EF 75085A MOV 0x08,#0x5A	
E-Svs		47: L_var2=0x5B;	
- 5,8	0v00	C:0x02F2 75095B MOV 0x09,#0x5B	
Ъ	0x00	48: L_var3=Dx5C;	
en	0x8f	C:UxU2F5 /SUASC MOV UxUA,#UxSC	
op dotr	0~000	49: L_Var4=UXI234; C.002E0 ZED12 MOV 0=0P #0=12	
PC \$	0x0000	$C \cdot 0 \times 0.2FB = 750G34 = MOV = 0 \times 007 \# 0 \times 34$	
ΓCΦ	0x0281	Solution in the second se	
⊡ psw	0.000	51:	*
			>
■ ■ \	🚨 🍕 F 🧒 T	Main.C 🕵 Disasse	

<u>5.2.2 原始码等级(Source-Level)的调试</u>

要做原始码等级的调试时,可以在 Files 分页中打开预调试的程序,再切回 Regs 分页即可返回寄存器窗口,如下 图所示:





5.2.3 设定断点

调试时最多可以同时设四个断点来使用。

插入/移除断点

将光标移至想要断点的指令上并按下右键,然后选"Insert/Remove Breakpoint"可以在该行指令做插入或是移除中 断,如下所示:

W test - 猩ision3 - [C:\tmp\test\Main.C]				7 🗙
Eile Edit View Project Debug Flash Peripherals Iools SVCS Window Help		2 Undo		×
Image: Second secon	xda	Redo Cut Copy Paste Select All Show Disassembly at 0xFF0002FE Set Program Counter Insert '#include <reg_mpc82g516.h></reg_mpc82g516.h>		
Image: System of the syste	may e	Run to Cursor line Go To Line Insert/Remove Breakpoint Enable/Disable Breakpoint Clear complete Code Coverage Info Outlining Advanced	•	- -

<u>启动/关闭断点</u> 将光标移至想要设定的指令上并按下右键,然后选"Enable/Disable Breakpoint"可以在该断点上设定是否启动或 是关闭该断点功能,当然,该行指令必须先插入断点。

W test - 猩ision3 - [C:\tmp\test\Main.C]	_	P	×
Eile Edit View Project Debug Flash Peripherals Iools SVCS Window Help		8	×
1 1 2 日 2 1 注 注 水 % % % %			
Project Workspace v x 042 unsigned char i; & Cut			
Register Value 043 D-Rege 044 //LED_blinking();			
10 0x00 045 L var1=0x5Å:			
r_1 r_2 r_2 r_3 r_4			
r3 Right clicking 046 L var9-0x1234; Show Disassembly at 0xFF0002FE			
r4 0x00 051 L_var3=0x3078; Set Program Counter			
r5 UXUU 052 L_var1=0x38; //! Note: this statement may Insert #include <reg_mpc82g516.h: r6 0x00 053 L_var1=0x77.</reg_mpc82g516.h: 	2		
17 0x00 054		-	
$ = -Sys \qquad \qquad 055 \qquad X_var1=0xA1; \qquad \qquad \ \ \ \ \ \ \ \ \ \ \ \ $			
b 0x00 057 058 fr war1=0x98: Go To Line			
sp Ox8f 059 G_var2=0xABCD; Insert/Remove Breakpoint			
PC \$ 0x0000 061 L_var1=G_var1;			
⊕ psw 0x00 062 X_var1=(char)G_var2; Clear complete Code Coverage Into 063			
064 for(i=0;i<16;i++) G array1[i]=i+0x60; Outlining			–
	•	É	
	_	-	



5.2.4 显示/编辑外围寄存器的内容

有许多的外围寄存器是不会显示在寄存器窗口的,要查看或是编辑这些寄存器必须在主选单中选择 Peripherals。 之后会有显示一个下拉窗口,用户可以自行勾选预查看的寄存器,如下所示:





5.2.5 检视反编译窗口

反编译窗口显示了<mark>原始码</mark>相对应该<mark>汇编语言</mark>,要开启这个窗口可以先在主选单中点选 View,之后会有显示一个下 拉窗口,再点选 Disassembly Window 即可检视反编译窗口,如下图所示:





5.2.6 检视查看窗口

检视查看窗口可以协助用户去查看区域变量以及全局变量,如下所示:

曜 test - 猩ision3 - [Disassembly]	
REile Edit View Project Debug Flash Peripherals	Iools SVCS Window Help
Carlor C	
Project Workspace 🗸 Debug Toolbar	31: void main(void)
Register Register r0 r0 r1 r2 Register Project Window Qutput Window r1 W Source Browser	32: Char L_var1,L_var2,L_var3; //local variables 34: int L_var4,L_var5; // 35: 36: Char xdata X_var1,X_var2; //local variables, in 'xdata' memory spa
r3 r3 r4 r3 r5 r4 r5 r5 r6 Memory Window r6 Memory Window r6 Memory Window r6 Memory Window r6 Symool r6 Symool r6 Symool r6 Symool r6 Symool sp Symool scrial Window optr Serial Serial Window #1 Scrial Serial Window #2 Serial	37: 38: LED_blinking(); 39: 0x02EF 120357 LCALL LED_blinking(C:0357) 40: L_var1=0x5A; 0x02F2 75085A MOV 0x08,#0x5A 41: L_var2=0x5B; 0x02F5 75095B MOV 0x09,#0x5B 42: L_var3=0x5C; 0x02F8 750A5C MOV 0x0A,#0x5C 43: L_var4=0x1234; 0x02FB 750B12 MOV 0x0B,#0x12 0x02FE 750C34 MOV 0x0C,#0x34 44: L_var5=0x5678; 45:
Symbols Toolbox	>
Mask: * Periodic Window Watch * Load "C: WS 1, `C Include File Der Window WS 1, `C Include File Der Window	Main.C Disasse ame Value L_var2 0x00 L_var2 0x00 L_var3 0x00 L_var4 0x0000 L_var5 0x000 X_var1 0x00 X_var2 0x00 X_var1 0x00 X_var2 0x00 X_var1 0x00 X_var2 0x00 X_var2 0x00 X_war2 0x00

要查看全局变量时要先选到 Watch #1 或 #2 分页, 然后按下<F2>键输入变量名称。

Project Workspace	~ ×	024 void test SFR(void);
Register	Value	025 void reset_SFR(void);
		026 Global variables
r0	0x00	027 unsigned char b_vari; //global variables
r1	0x00	D20 unsigned int b_valz, //
r2	0x00	0.50 //***********************************
r3	0x00	031 void main(void)
r4	0x00	
r5	0x00	033 char L_varl,L_var2,L_var3; //local variables
гб	0x00	
r7	0x00	036 char xdata X var1.X var2: //local variables, in 'xdata' memory space
⊟ Svs		037
a	0x00	E2038 LED_blinking();
Ъ	0x00	039 J. max1=0x54
SD	0x13	1040 L_vali=0.5.8; 1041 L_var2=0.5.8;
dotr	0x0000	042 L var3=0x5C:
PC \$	0x02ef	043 $L_var4=0x1/234;$
± psw	0x00	044 L_var5=0x9678;
0.4070700	0.56.002	045 U vert-0.82. // Note: this statement were be entirized out
	<u>ОЛ 🕴 г. 🔍 т. </u>	
		E Main.C
× Load "C:\	\tmp\\test\\test'	Value Address: w0w0000
'WS 1, `G_`	varl	- ('G var 1 0x00
WS 1, `G_`	var2	G yar2 0x0000 X:0x000000: 00 00 DF E0 6B 71 77 FD -
		Stype F2 to edit
		Press $\langle F^2 \rangle$ key to $0 \neq 0.00010$: DC E7 39 B5 76 97 C9 14
		enter clobal variable 0x000020: F6 4F 30 DF 8F FA AA 5B
> dev		0x000028: CD 19 39 CF F7 1B F9 49
ASM ASSIG	N BreakDisable	
		The second watch #1 watch to the second seco
8	Nedi	Since a second s



5.2.7 检视内存窗口

要打开这个窗口,可以在主选单中点选 View,之后会有显示一个下拉窗口,再点选 Memory Window,如下图所示,而这个窗口支持的命令有四种:

- (1) 检视'data'内存: d:0x00~d:0xFF
- (2) 检视'idata'内存: i:0x00~i:0xFF
- (3) 检视'xdata'内存: x:0x0000~x:0xFFFF
- (4) 检视'code'内存: c:0x0000~c:0xFFFF

使用者可以用相对应该的命令查看这四种内存的内容,如要查看外部记忆'xdata'的内容可以参考第 6.2 节

🕎 test - 猩ision3 - [Disassembl y]	
REile Edit View Project Debug Flash Peripherals	Tools SVCS Window Help
The Status Bar	≥ ∞ ∞ ∞ ∞ ∞ ∞ ∞ ∞ ∞
Ref 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
Project Workspace 🗸 Debug Toolbar	31: void main(void)
Register	33: char L_var1,L_var2,L_var3; //local variables
🕺 🔽 Qutput Window	34: int L_var4,L_var5; // 35:
r2 Source Browser	36: char xdata X_var1,X_var2; //local variables, in 'xdata' memory spa
13 Disassembly Window	38: LED_blinking();
r5 Watch & Call Stack Window	0x02EF 120357 LCALL LED_blinking(C:0357)
r6 🔲 Memory Window	40: L_var1=0x5A; 0x02F2 75085A MOV 0x08,#0x5A
Sys E Performance Analyzer Window	11: L_var2=0x5B; 0x0285 75095B MOV 0x09 #0x5B
b Logic Analyzer Window	42: L_var3=0x5C;
sp Is Symbol Window	43: L_Var4=0x1234; i:0x00~i:0xFF for 'data' type
dptr Serial Window #1	0x02FB 750B12 MOV 0x0B,#0x12 0x02FE 750C34 MOV 0x0C,#0x34 x:0x0000~x:0xFFFF for 'xdata' type
Files Serial Window #2	44: L_var5=0x5678; c:0x0000~c:0xFFFF for 'code' type
Symbols 🅕 Icolbox	43.
Mask: * Poriedie Window Undate	Main.C R Disasse Window
× Load "C	ame Value Address: x-0x0000
WS 1. C Include File Dependencies	L_var1 0x00 X:0x000000: 00 00 DF E0 6B 71 77 FD
	L_var2 0x00 X:0x000008: BB 1A 3C D7 AE 73 5C 38 -L_var3 0x00 X:0x000010, DC F7 39 B5 F6 9F C9 14
	L_var4 0x0000 X:0x000018: D2 ED AF EF FA 1D 31 C7
No contraction of the second s	L_val 0x0000 X:0x000020: F6 4F 30 DE 8E EA AA 5B -X_varl 0x00 X:0x000028: CD 19 39 CF F7 1B F9 49
ASM ASSIGN BreakDisable	<u>−X_var2</u> 0x00 X:0x000030: D7 AB D4 65 7F DB 77 B7 X:0x000038: 5F 6F 18 B9 A5 7E B5 5F ▼
	Locals & Watch #1 & Watch #2 E M Memory #1 & Memory #2 & Memory #
Memory Window	Merawin On-Chin-Debug Driver t1:0.00000000 sec



6 注意事项

<u>6.1 寄存器定义文件</u>

寄存器定义文件REG_MPC82G516.INC及REG_MPC82G516.H定义了所有的特殊功能寄存器(SFRs)以及可位寻址的位。在安装OCD ICE时(見第二节)会同时将此寄存器定义文件安装至Keil 8051 IDE的预设目录内,因此,当使用Keil编写程序时可以直接使用\$INCLUDE (REG_MPC82G516.INC)或是#include <REG_MPC82G516.H>来将定义文件(头文件)引入,而不需要将定义文件拷贝到您的项目目录。

6.2 内建 XRAM 及外部数据存储器

Megawin 8051 提供了内建的内存 XRAM (eXpanded RAM),他的存取方式就如同一般传统的外接内存一样,MPC82G516 的内建内存容量为 1024 个字节,地址是从 0x0000 到 0x03FF,而由于这个内建内存的地址会跟外部内存的地址重叠到,因此必须要有一个控制位来区分这两个物理内存的地址。此时,这个 ERAM(暂存器 AUXR 的 bit-1)位扮演了这个角色。因为 C51 的组译器不会自动去帮使用者切换这两个内存,用户要使用内建 XRAM 时必须手动去清除这个位,要使用外接内存时必须手动去设定这个位,而这个位一开机或是 Reset 后的默认值为 0。

C51 的组译器提供了两种不同存取外部内存的方式: xdata 以及 pdata (xdata 可以定位到 64K 位的外部内存数据, 而 pdata 尽能定位到 256 位的数据), 当用户想要直接在内存窗口检视 xdata 或 pdata 而不是在查看窗口时,则必须选取主选单的 Peripherals- XRAM 再选择"Display xdata from on-chip XRAM"或是"Display xdata from external RAM"来选择 要检视内建 XRAM 或是外接内存,操作方式如下图所示:

🕎 test - 猩ision3 - [C:\tmp\test\Main.C]



来查看G_array1[]阵列,用显示外接内存"Display xdata from external RAM"来查看 G_array2[]阵列。

a 🗙



同时使用内建及外部内存的范例

```
unsigned char xdata G_array1[512] _at_ 0x0000; // in 'xdata' space, will use on-chip XRAM unsigned char xdata G_array2[512] _at_ 0x0000; // in 'xdata' space, will use ext. RAM unsigned int i;
```

AUXR&=0xFD; // clear AUXR.1 for on-chip XRAM for (i=0; i<512; i++) G_array1[i]=0x5A; // fill XRAM with 0x5A

AUXR|=0x02; //set AUXR.1 for external RAM for (i=0; i<512; i++) G_array2[i]=0xA5; // fill ext. RAM with 0xA5

在组译时会出现以下的警告讯息,然而这是没关系的,因为我们故意将 G_array1 及 G_array2 定义成相同的位 址,但事实上,我们使用了 ERAM 这个位切换不同的物理内存空间。

linking...

- *** WARNING L6: XDATA SPACE MEMORY OVERLAP FROM: 0000H
 - TO: 01FFH

6.3 程序代码优化及原始码调试

在以下的原始码中,**C51**的组译器将不会产生"L_var1=0x38"的机械码,因为下一道指令为"L_var1=0xC7",因此这道指令将会变成没有意义的,所以必须将程序代码优化关闭才不会将这道指令忽略掉,如4.4节的动作

unsigned char L_var1;

L_var1=0x38; // ! Note: this statement may be optimized out by the C51 compiler L_var1=0xC7;

所以,当执行原始码调试,执行到这道指令时 L_var1 将不会显示 0x38 而可能显示一个乱数,事实上,这道指令并 没有对应的机器码,使用者必须注意到这一点!

有的时后,为了要调试,使用者会将程序代码优化关闭,此时可能会出现开启时不会出现的连接错误。例如下图的错误讯息,他的意思是你的变量超出了 MCU 内存的范围,要让这个错误消失,唯一的方式是开启优化让组译程序更有效率的使用内存。

linking...

*** ERROR L107: ADDRESS SPACE OVERFLOW SPACE: DATA SEGMENT: ?DT?_VP_DISPLAYMODE?VP LENGTH: 0001H



6.4 for 循环的原始码调试

以下两组指令对于 8051 的 CPU 是完全一样的,当用单步原始码调试时,在第一组指令是没有问题的,然而,如 果在第二组指令时将会花上许多的时间,我们认为那是因为在 Keil 的调试功能里这样的程序是未知的,在我们尚未 得到 Keil 的回复之前,我们建议尽量使用第一组指令取代第二组指令,如果要使用单步执行来执行这类指令时。要 调试第二组指令的另一个方式是将游标移至第二行并执行 Run-to-Cursor 键来跳过第一行。

指令 1: Line1: for (i=0; i<16; i++) { Line2: G_array1[i]=i+0x60; Line3: }

指令 2:

Line1: for (i=0; i<16; i++) G_array1[i]=i+0x60; Line2: ... Line3: ...

6.5 使用调试时的硬件选项要求

在 dScope-Debugger 模式下有两个硬件选项的需求:

需求一:调试的芯片必须在没有上锁的狀态

因为当要调试的芯片上锁的时后,在 dScope 模式下要下载用户的应用程序到芯片之前会将芯片做完全删除,因此所有的硬件选项将会被关闭,所以有可能原本设定的硬件选项会遗失而照成动作不正常。例如,有一颗有设定 IAP 的芯片上锁了,在进入 dScope-Debugger 模式并下载程序之后,IAP 的设定将会消失,所以芯片执行就会出问题了。

需求二:调试的芯片必须将 ISP 的功能关闭

由于当 ISP 功能开启时,这个调试的芯片每次重开机都会从 ISP 的地址开始执行并且执行 ISP 的程序,所以当进 入 dScope-Debugger 模式时会给芯片一个 Reset 的命令,此时将会执行在 ISP 地址内的程序(例 ISP-code),而不 是执行 Keil 所开启的项目,所以要做调试功能时,必须将 HWBS 关闭以防止执行到 ISP 的功能。

注:

在应用程序调试完成后,使用者可以用 "Megawin 8051 ICP Programmer"来存回原来的硬件配置



<u>6.6 错误讯息</u>

当有下列情况时会显示错误讯息"Error: Target DLL has been cancelled. Debugger aborted !"如下图所示:

- (1) ICE 转接器故障(可重启电脑尝试)
- (2) 主板芯片没有动作或目标MCU没有电源供应(例如没开电源或损毁)
- (3) 连接 ICE 及 MCU 的连线坏掉、接触不良或是接线错误
- (4) KEIL开发环境没有设置好,或没有新增Megawin芯片的资料到Keil 8051 IDE,解决办法请按3.2节安装好
- (5) 以上方法不能找出原因时,请打开"IcpProgrammer.exe"直接下载一个 hex 档到 MPC82G516,以验证 ICE 工具是否损坏.

当有此错误讯息时,按下 "OK"然后看看是不是有以上情形以解决这个问题。





6.7 正确的连接 ICE 仿真器到计算机

如果计算机先经由一个 USB 分享器再接到 ICE 仿真器的话,数据传输速度将会被大大的减低,所以如果要用 dScope 的功能进行调试的话,使用者必须直接将 ICE 仿真器直接接到计算机上才行,如图 6.7.1,而不要经过分享 器再到计算机,如图 6.7.2

图 6.7.1 直接接到计算机的 USB 端口



图 6.7.2 不要经过分享器再接到计算机的 USB 端口





修订记录

Revision	Description	
v1.00	The first release for beta-site test.	2007/08/15
v1.01	Add notes when installation fails. (Section 3.2)	2007/08/24
v1.02	Change to manually specify the installation path of the Keil software. (Section 3.2)	2007/08/27
	Add the notification of default installation path of Keil 8051 IDE software. (Section 3.2)	2007/08/29
v2.00	Update the Keil IDE Setup. (Section 4.4)	2007/10/08
	Update the Special Notes. (Section 6)	
	The formal released version.	2007/10/08
v2.10	(1) Improve the defect of breakpoint setting.(2) Fix the bug of wrong erasing range when downloading the application code.	2007/12/26
V2.20	(1) Update the data base for all series of MCU in Driver Installer.(2) Removed the function of detecting the ICE adapter when install Driver.	2009/02/27
v2.21	Change the folder name of Driver Install to Database install	2009/04/01
v2.30	(1) Supported MG82FL(E)532 and MG82FL(E)564(2) Supported ICP function	2010/05/10
v2.31	Update "Database Installer "	2010/05/21
v2.32	Support uVision4	2010/06/02
v2.33	Update "IcpProgrammer.exe" in Database Installer	2010/08/25
V2.40	(1) Supported MA806-24, MA806-32 and MA806-64(2) Supported MG84FG516	2011/02/11